Lost in Quantization: Improving Particular Object Retrieval in Large Scale Image Databases

James Philbin[1], Ondřej Chum[2], Michael Isard[3], Josef Sivic[4], Andrew Zisserman[1]

[1] Visual Geometry Group, Department of Engineering Science, University of Oxford
[2]Center for Machine Perception, Faculty of Electrical Engineering, Czech Technical University in Prague
[3]Microsoft Research, Silicon Valley
[4]INRIA, WILLOW Project-Team, Laboratoire d'Informatique de l'Ecole Normale Superieure, Paris, France

Abstract

The state of the art in visual object retrieval from large databases is achieved by systems that are inspired by text retrieval. A key component of these approaches is that local regions of images are characterized using high-dimensional descriptors which are then mapped to "visual words" selected from a discrete vocabulary.

This paper explores techniques to map each visual region to a weighted set of words, allowing the inclusion of features which were lost in the quantization stage of previous systems. The set of visual words is obtained by selecting words based on proximity in descriptor space. We describe how this representation may be incorporated into a standard tf-idf architecture, and how spatial verification is modified in the case of this soft-assignment.

We evaluate our method on the standard Oxford Buildings dataset, and introduce a new dataset for evaluation. Our results exceed the current state of the art retrieval performance on these datasets, particularly on queries with poor initial recall where techniques like query expansion suffer. Overall we show that soft-assignment is always beneficial for retrieval with large vocabularies, at a cost of increased storage requirements for the index.

1. Introduction

We are interested in the problem of specific object retrieval from an image database. In other words, given a query image in which a particular object has been selected, our system should return from its corpus a set of representative images in which that object appears. This is a harder problem than whole-image retrieval, since the query object may be occluded, lit differently, or seen from different viewpoints in returned images. On the other hand it is in many ways simpler, and better specified, than the related problem of object category retrieval, which requires some abstraction of the common visual appearance of all objects within a given category.

Several successful object retrieval systems have recently appeared [7, 9, 14, 15], using approaches inspired by the text retrieval literature in the manner of [17]. A key component of these approaches (which are reviewed in more detail in section 2) is that local regions of images are characterized using "visual words" selected from a discrete vocabulary. The function that maps a high-dimensional region descriptor into this vocabulary is an active area of research, but the most successful approaches all perform some form of clustering or quantization using example images as a training set.

In this paper, we build on previous work that trains its vocabulary using a small set of representative images. Substantial engineering effort has been devoted in recent years to the study of feature detection, summarizing image regions using invariant descriptors, and clustering these descriptors, and we adopt state of the art methods for these tasks. The novelty of our work is in the use that we make of the clustered descriptors. Recent work [7, 15] has shown that these methods can suffer from poor recall: feature detectors often fail to fire even on near-duplicate images, and query regions often fail to contain the visual words needed to retrieve matches from the database. One very successful technique for boosting recall is query expansion [7] which achieves substantially better retrieval performance when the visual words in a query region are augmented using words taken from matching regions in the initial results set. However, this method relies on sufficient recall from the initial query to get the process started, and can fail badly on queries with poor initial recall.

Our approach, described in section 3, specifically addresses the problem of recall from an initial query, and is therefore complementary to query expansion methods. It relies on "soft-assignment," so that a high-dimensional descriptor is mapped to a weighted combination of visual words, rather than "hard-assigned" to a single word as in previous work. Thus we address the problem of failing to retrieve image patches whose descriptors have been "lost in quantization".

The paper mainly concentrates on a mechanism we call "descriptor-space soft assignment," which generates a single descriptor for each patch as is usual, but then associates that descriptor with r nearby cluster centers instead of its single nearest-neighbour cluster. This can be thought of as very loosely analogous to "stemming" in text retrieval where a search term is expanded to include textually similar variants. We also briefly discuss experiments using "image-space soft assignment" where we generate multiple descriptors by perturbing the image patch directly, and choose visual words to approximate the space of these perturbed descriptors. The function from images to descriptors is not continuous, so this can result in descriptors which are distant. This technique bears some resemblance to synonym/acronym expansion in text retrieval, where an acronym such as "CIA" might be expanded to the search terms "Central Intelligence Agency."

Section 4 explains how the soft-assigned words are used in a retrieval system. Section 5 describes the datasets we use to evaluate the approach, and section 6 describe the experiments we have performed. We show results both when the underlying vocabulary is trained on the corpus images, and also when training is performed on similar but distinct scenes. We report substantial improvement to the baseline using soft assignment, particularly when the vocabulary is not trained on the corpus images. In section 7 we examine the performance of image-based soft assignment in comparison to the descriptor based method. Section 8 concludes with a discussion.

2. State of the Art

This section overviews the state of the art of real-time large image corpus object retrieval. Most of the successful engines are based on the bag-of-visual-words approach [7, 9, 14, 15, 17].

Image description. For each image in the dataset affine invariant interest regions are detected. Popular choices are MSER [12, 14] or multi-scale Hessian interest points [13, 15]. Each detected feature determines an affine covariant measurement region, typically an ellipse defined by the second moment matrix of the region. An affine invariant descriptor is then extracted from the measurement regions. Often a 128-dimensional SIFT [11] descriptor is used.

Quantization. Vector quantization of feature descriptors for object retrieval was originally suggested in [17]. In that work, small vocabularies of 10K and 6K clusters were generated using k-means. The time complexity of the k-means algorithm is $O(kN)$, where N is the number of data points. Such a time complexity is feasible for small values of k, but renders the algorithm intractable for large vocabularies ($k > 10^5$).

It was shown [14, 15] that for large scale image / object retrieval a more discriminative vocabulary is necessary. The time complexity is, in the case of [14], reduced by exploiting a nested structure of Voronoi cells, known as Hierarchical k-means (HKM) [8]. Instead of solving one clustering with a large number of cluster centers, a tree organized hierarchy of smaller clustering problems is solved. This reduces the time complexity to $O(N \log k)$. In [15] it was shown that this reduced time complexity could also be achieved by replacing the nearest neighbour search of k-means by a KD-forest approximation [11, 16]. The experiments of [15] demonstrated that vector quantization obtained by this Approximate k-means (AKM) is superior to HKM. A fixed quantization method (complexity $O(N)$) was suggested in [18].

Search engine. The search engines used for image / particular object search have been inspired by widely used text search engines [3, 5]. Such a search engine uses the vector-space model of information-retrieval. The query and each document in the corpus is represented as a sparse vector of term (visual word) occurrences and search then proceeds by calculating the similarity between the query vector and each document vector. The standard tf-idf weighting scheme [4] is used, which down-weights the contribution that commonly occurring, and therefore less discriminative, words make to the relevance score.

For computational speed, the engine stores word occurrences in an index, which maps individual words to the documents in which they occur. For sparse queries, this can result in a substantial speedup over examining every document vector, as only documents which contain common (to the query) words need to be examined. The scores for each document are accumulated so that they are identical to explicitly computing the similarity.

Spatial verification. Up to this point, the bag-of-visual-words based retrieval was discussed and all the spatial information was ignored. As shown in [7, 15, 17], the results can be significantly improved using the feature layout to verify the consistency of the retrieved images with the query region.

The initially returned result list is re-ranked by estimating affine homographies between the query image and each of the top-ranking results from the initial query. The score used in re-ranking is computed from the number of verified inliers for each result.

Contextual dissimilarity measure. Typically, the dissimilarity between the (appropriately normalized) query and image visual word vectors is measured by the L_1 or L_2 distance [7, 14, 15, 17]. These standard dissimilarity measures could be further modified to depend on the local density around each image vector in the visual word vector space, essentially "pushing" images in the densely populated areas of the vector space away from the query. This "contextual

dissimilarity measure" [9] was found to improve retrieval performance on the image retrieval benchmark of [14], but requires computing k-nearest neighbouring images for each image in the database, which may become prohibitively expensive on very large image collections.

Query expansion. In the text retrieval literature, a standard method for improving performance is query expansion, where a number of the highly ranked documents from the original query are reissued as a new query. This allows the retrieval system to use relevant terms not present in the original query.

In [7], query expansion was brought into the visual domain. A strong spatial constraint between the query image and each result allows for an accurate verification of each return, suppressing the false positives which typically ruin text-based query expansion. These verified images can then be used to learn a latent feature model to enable controlled construction of expanded queries.

The simplest well performing query expansion method is called average query expansion. A new query is constructed by averaging a number of document descriptors. The documents used for the expanded query are taken from the top verified results of the original query.

2.1. The baseline

For a baseline retrieval system, we follow the architecture of our previous work [15]. We detect Hessian interest points and fit affine covariant ellipses [13]. On average, there are 3,300 regions detected on an image of size 1024×768. For each of these affine regions, we compute a 128-dimensional SIFT descriptor [11].

A visual vocabulary of 1M words is generated using an approximate k-means clustering method [15]. Each visual descriptor is assigned, via approximate nearest neighbour search, to a single cluster center, giving a standard bag-of-visual-words model. These quantized visual features are then used to index the images for the search engine.

To reach the state of the art results, we employ the query expansion method of [7], using the average expansion method.

3. Soft Assignment of Visual Words

In the bag-of-visual-words representation two image features are considered identical if they are assigned to the same visual word (cluster center). On the other hand, two features assigned to different (even very close) clusters are considered totally different. In effect the quantization provides a very coarse approximation to the actual distance between the two features — zero if assigned to the same visual word, and infinite otherwise. In practice this hard assignment leads to errors because of variability in the feature descriptor.

This variability arises from many sources: image noise, varying scene illumination, instability in the feature detection process and non-affine changes in the measurement regions. Distortions that cannot be handled by the invariance built into the descriptor result in a change in the descriptor value, and in turn this may result in the same surface patch being assigned to different visual words in different images. Typically, descriptors corresponding to the same physical patch in different images will be close together, however this is not always the case. Severe image distortions, such as strong lighting variations (*e.g.* cast shadows) or self occlusions can abruptly change the descriptor value of the patch. In many cases that are of our interest, even a small change in the patch appearance can change the descriptor dramatically. These cases are related to errors in establishing an invariant coordinate system (*e.g.* during dominant orientation detection in the SIFT descriptor).

Our objective therefore is to describe an image of a surface patch by a weighted combination of visual words, such that there is an improvement in matching (compared to a hard assignment) of the surface patch between images. In this paper we describe two different approaches to choosing this weighted combination. In the first approach, referred to as "descriptor-space soft-assignment", we extract a single descriptor from each image patch and assign it to several visual words nearby in the descriptor space. In section 7 we briefly discuss a second approach, referred to as "image-space soft-assignment", in which we extract a *set* of descriptors from each image patch by synthesizing deformations of the patch in the image space and assign each descriptor to the nearest visual word.

3.1. Descriptor-space soft assignment

The term "soft assignment" is commonly used [6] in histogram comparisons. It describes techniques that identify a continuous value with a weighted combination of nearby bins, or "smooth" a histogram so that the count in one bin is spread to neighbouring bins.

In this section we investigate a descriptor dependent soft-assignment where the weight assigned to neighbouring cells depends on the distance between the descriptor and the cell centers. The intuition is that the weight vector can act as a local coordinate frame to more precisely localize the descriptor in SIFT space. The benefit of such soft-assignments are illustrated in figure 1.

As is usual in soft-assignment (for example in estimating Gaussian Mixture Models [6]) the weight assigned to a cell is an exponential function of the distance to the cluster center. We assign weights to each cell proportional to $\exp -\frac{d^2}{2\sigma^2}$, where d is the distance from the cluster center to the descriptor point. In practice σ is chosen so that a substantial weight is only assigned to a small number of cells. The essential parameters are then: the spatial scale σ and

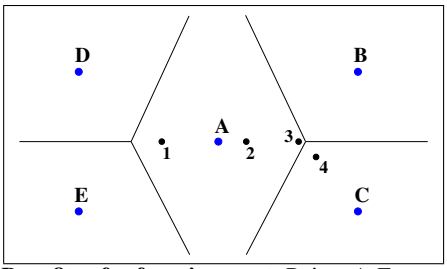

Figure 1. **Benefits of soft assignment.** Points A-E represent cluster centers (visual words), and points 1–4 are features. Here we demonstrate two benefits of soft assignment: (i) In hard assignment, features 3 and 4 will never be matched as they are assigned to different visual words despite being close in descriptor space. Using soft-assignment, words 3 and 4 will be assigned to A, B and C (with certain weights) and can be matched strongly as they are close in the descriptor space; (ii) In hard assignment, features 1–3 are all assigned to word A equally and there is no way of distinguishing that 2 and 3 are closer than 1 and 3. Soft-assignment provides a way of recording this information, and subsequently giving more weight to the closer matches and less to the further.

the number, r, of nearest neighbours considered. Note that after computing the weights to the r nearest neighbours, the descriptor is represented by an r-vector, which is then L_1 normalized. In section 6, we give experimental result for several settings of values for σ and r.

The descriptor-space soft assignment "plugs into" the standard index architecture of the search engine at a cost of more storage (since the index is less sparse) but limited extra run time complexity at the index stage. There is a cost at the spatial verification stage however.

4. Large scale object retrieval with soft visual words

Changing the representation of a visual feature from a single word to an r-vector affects elements of the retrieval system that have until now assumed a single word id per feature. We now describe the changes required in the search engine.

4.1. TF–IDF weighting and soft assignment

The *tf-idf* weighting scheme is generally applied only to integer counts of visual-words in images. It requires some modification to handle a descriptor represented by a weight r-vector. We adapt this weighting scheme for soft clustering as follows. For the term frequency we simply use the normalized weight value for each visual word. For the inverse document feature measure, we found that counting an occurrence of a visual word as one, no matter how small its weight, gave the best results.

4.2. Spatial re-ranking and soft assignment

When advancing from hard to soft assignment, one of the major performance considerations is the potential for growth in the number of *tentative* correspondences between two images (defined as the set of features which share at least one visual word assignment). This growth is because one feature in an image has more assigned visual words and can potentially match more features in the other image.

However, since our visual vocabulary is specific and large (1M), the probability that two unrelated features are assigned to the same visual word is small. Therefore, we empirically observe a roughly linear growth in number of tentative correspondences as the number of nearest neighbours taken increases. For $r = 3$, we find that the average increase in tentative correspondences is 3.24.

The set of tentative correspondences that are consistent with an affine homography is determined by a RANSAC style estimation, as described in [15]. This requires a scoring for each hypothesised transformation, and the scoring function can make use of the weighted vector associated with each feature, rather than simply counting the number of inlier correspondences.

Suppose features x and y are matched and their weight r-vectors are $\mathbf{w}_x$ and $\mathbf{w}_y$ respectively, then for example the scoring function could be the scalar product $\mathbf{w}_x.\mathbf{w}_y$, or the (cosine of the) angle between the vectors $\mathbf{w}_x.\mathbf{w}_y/||\mathbf{w}_x||||\mathbf{w}_y||$ (since $\mathbf{w}_x$ and $\mathbf{w}_y$ are L_1, not L_2, normalized). The score for the hypothesis is simply taken as the sum of the scores for all the inliers. We give performance results over several scoring functions in section 6.

5. Datasets and Evaluation

To evaluate our system, we use the *Oxford Buildings* dataset available from [1]. This is a relatively small set of 5K images with an extensive associated ground truth for 55 standard queries: 5 queries for each of 11 Oxford landmarks. To examine the generalization of our system when the testing dataset (here Oxford) differs from the dataset used to generate the quantization (visual words), we have collected a new training dataset of a different city – Paris. We also use an additional unlabeled dataset, *Flickr1* which is assumed not to contain images of the ground truth landmarks. The images in these additional datasets are used as "distractors" for the system and provide an important test for the scalability of our method. These three datasets are described below and compared in table 1. The set of images downloaded from two or more of Flickr's tags will not in general be disjoint, so we remove exact duplicate images from all our datasets.

The *Oxford* dataset. This dataset [1] of 5,062 images is a standard particular object retrieval test set [7, 15]. A sample of 4 query images is shown in figure 2, for the rest see [1].

Figure 2. Some of the 55 Oxford query images.

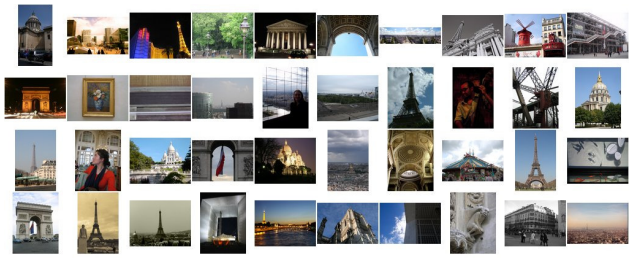

Figure 3. A random sample of images from the Paris dataset.

Dataset	Number of images	Number of features
Oxford	5,062	16,334,970
Paris	6,300	20,219,488
Flickr1	99,782	277,770,833
Total	111,144	314,325,291

Table 1. The number of descriptors for each dataset.

The *Paris* dataset. This dataset, used for training the SIFT descriptor quantizers in some experiments in this paper, contains 6,300 high resolution (1024×768) images obtained from Flickr by querying the associated text tags for famous Paris landmarks such as "Paris Eiffel Tower" or "Paris Triomphe." Example images from this dataset are shown in figure 3. The motivation for choosing Paris landmarks is to have images of similar scenes to those of the Oxford landmarks (i.e. buildings, often with some similarities in architectural style), but without having identical buildings to those used in the Oxford quantization. More details for this dataset can be found in [2].

***Flickr1* dataset.** This dataset was crawled from Flickr's 145 most popular tags and consists of 99,782 high resolution images.

5.1. Evaluation procedure

To evaluate performance we use Average Precision (AP) computed as the area under the precision-recall curve. Precision is the number of retrieved positive images relative to the total number of images retrieved. Recall is the number of retrieved positive images relative to the total number of positives in the corpus. An ideal precision-recall curve has precision 1 over all recall levels, which corresponds to an Average Precision of 1.

An Average Precision score is computed for each of the 5 queries for a landmark specified in the Oxford Buildings dataset, and these are averaged to obtain a Mean Average Precision (mAP) for the landmark. For some experiments, in addition to the mAP, we also display precision-recall curves which can sometimes better illustrate the success of our system in improving recall.

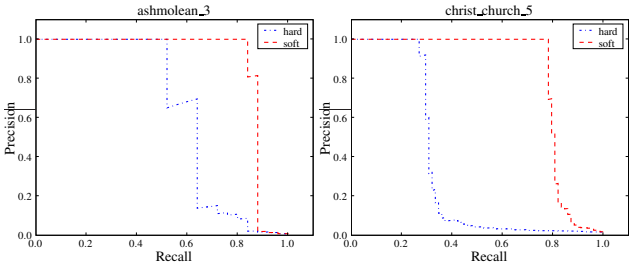

Figure 4. Some examples of the improvement in mAP obtained by using soft-assignment on two different queries, with spatial and query-expansion turned on. The results for hard-assignment is shown in blue with the descriptor soft-assignment method shown in red.

We evaluate our system on two databases – D1 composed of only the *Oxford* dataset (5,062 images), and D1 + D2 composed of *Oxford* (D1) + *Flickr1* (D2) datasets The combined datasets of *Oxford* and *Flickr1*, consists of 104,844 images. The vector quantizers are trained on either the *Oxford* or *Paris* datasets. The effect of the training data used for quantization and the size of the image database on the performance is discussed in section 6.

6. Experimental evaluation

The goal here is to evaluate the benefits of descriptor based soft-assignment. In particular, we test the sensitivity of soft-assignment to different parameter settings, compare performance to alternative quantization methods, and evaluate the benefit of soft-assignment when combined with spatial re-ranking and query expansion. Unless otherwise stated the performance is measured on the D1 dataset consisting of 5K Oxford images.

Parameter variation: Table 2 shows the retrieval performance with different soft-assignment parameter settings as evaluated on the Oxford dataset, trained using both Oxford and Paris. Here we test only the bag-of-visual-words retrieval with tf-idf weighting as described in section 4.1, i.e. no spatial verification or query expansion is used.

It can be seen from the table that the performance of the system as these two parameters are varied changes very little. In particular, soft-assigning to more than 4 nearest neighbours doesn't bring any additional benefit, which might be attributed to increased confusion during matching. Therefore, as a compromise between retrieval quality and extra computational cost and memory requirements, we use r=3, σ^2=6,250 in all subsequent experiments.

Comparison with other methods: Here we compare the bag-of-visual-words retrieval (*i.e.* no spatial ranking or query expansion) using the soft- and hard-assigned vocabularies with our implementation of hierarchical k-means (HKM) of [14] and fixed quantization of [18]. In the case of

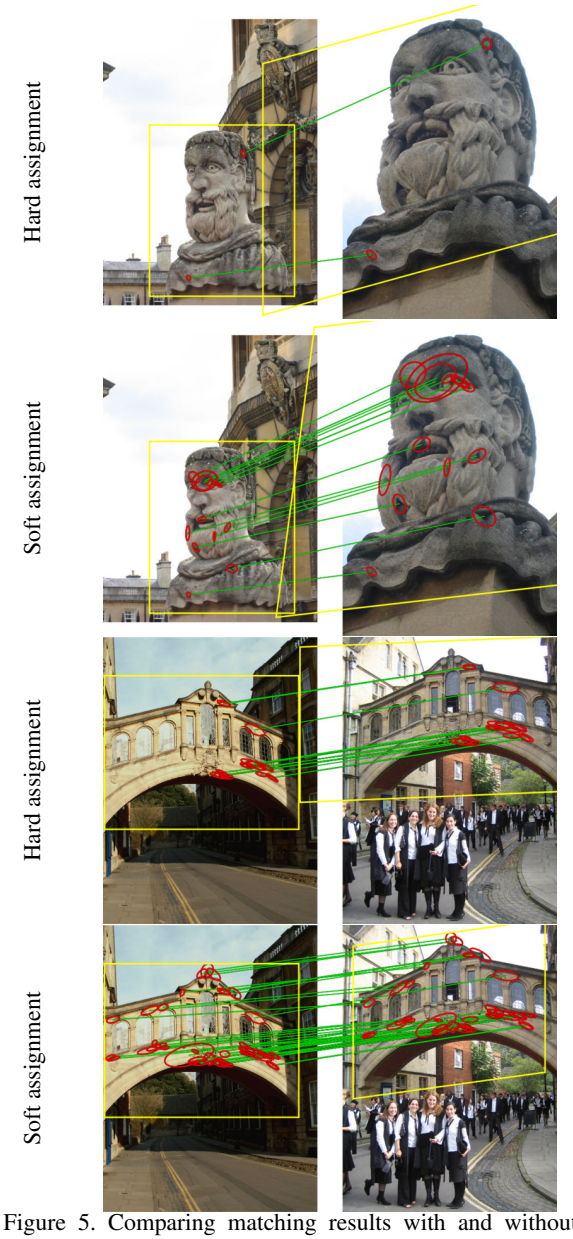

Figure 5. Comparing matching results with and without soft-assignment. The query image and region is shown on the left with the results of matching after spatial verification shown to the right. When soft-assignment is used, many more inliers are found and the object is better localized in the target image.

HKM we used the L_2 distance with 1 and 2 levels of hierarchical scoring (L_1 was found to degrade performance on this dataset [15]). Results are summarized in table 3. The soft-assigned vocabulary performs the best when trained either on the *Oxford* or *Paris* datasets. Note however, the drop in performance (consistent for all methods) when the vocabulary is trained on images of a different city (Paris) than the test dataset (Oxford).

The fixed quantization method [18] performs significantly worse as it is not suitable for specific object retrieval.

r	σ^2	Training data	
		Oxford	Paris
3	5,000	0.671	0.495
3	6,250	0.673	0.494
3	7,500	0.672	0.493
5	5,000	0.674	0.502
5	6,250	0.673	0.499
5	7,500	0.673	0.496

Table 2. Comparison of different parameter settings for the descriptor based soft-assignment on the *Oxford* dataset. These results are for a 1M vocabulary, tested on D1.

Method	Training data	
	Oxford	Paris
Fixed Quantization [18]	0.164	
HKM [14] (1 level)	0.422	0.401
HKM [14] (2 level)	0.410	0.340
Hard [15]	0.614	0.403
Soft	**0.673**	**0.494**

Table 3. Comparison of soft- and hard-assigned vocabularies with the hierarchical k-means and fixed quantization. The performance is evaluated on the *Oxford* dataset. Except for the fixed quantization method (which does allow for altering vocabulary size) all the methods used 1M visual words.

We believe that this method splits dense regions of the descriptor space arbitrarily along dimension axes, and the bins do not equally split the unit hypersphere which SIFT covers, resulting in a wildly uneven distribution of points.

Effect of vocabulary size: We now evaluate the efficacy of soft-assignment for different vocabulary sizes. The results are shown in figure 6. It can be seen that soft-assignment produces a much greater benefit when larger vocabularies are used. We attribute this performance boost to the ability of soft-assignment to overcome some over-quantization of the space when large vocabularies are used.

We employ high levels of compression within our index (which stores the bag-of-visual-words histograms) to reduce runtime memory usage. Unfortunately soft assignment seems to reduce the predictability of the data and slightly lowers our compression ratios. For the D1 dataset with 1M vocabulary and hard assignment, the index was 36MB. Using the same data but with soft assignment to 3-NN, the size of the index increased to 108MB.

6.1. Spatial re-ranking

In this section, we examine the performance of the system when spatial re-ranking and query expansion is used with soft-assignment.

Spatial verification: Here, we experiment with several different functions for scoring hypotheses in the spatial re-ranking stage. A candidate hypothesis is scored by summing a score computed from the weights of the two features partially assigned to the same word. We find that this allows

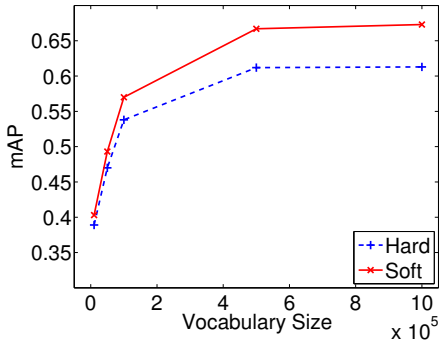

Figure 6. Comparing the performance of hard assignment and soft assignment for differing vocabulary sizes, training and testing on D1.

Normalization, n	Similarity, s	mAP
No spatial		0.673
L_1	Dot product	0.731
L_1	Histogram intersection	0.722
L_2	Dot product	0.730

Table 4. Comparing different scoring methods for spatial verification. These results are for a 1M vocabulary, trained on D1, tested on D1.

the verification to use more visual information at matching time and gives improved robustness in challenging scenes.

Table 4 shows the results of using different similarity measures and normalizations, for scoring functions. The score of a hypothesis is computed as $\sum s(n(\mathbf{w}_x), n(\mathbf{w}_y))$ where the sum is over the inliers, s is the similarity measure, and n the normalization of the r-vectors for a correspondence between features x and y. It can be seen that changing the similarity and normalization has little effect on the overall mAP. From now on we use an L_1 normalization with a dot product similarity.

Query expansion: The goal here is to test whether the soft-assigned vocabulary can be effectively combined with spatial verification (section 4.2) and query expansion (section 2). Results are summarized in table 5. The soft-assigned vocabulary consistently outperforms the hard-assigned vocabulary even when combined with spatial verification and query expansion (from 0.801 to 0.825, i.e. a reduction in error by 12.1%). The benefit of soft-assignment is even more obvious when the vocabulary is trained on the Paris dataset (from 0.654 to 0.718, a reduction in error by 18.5%). This suggests that soft-assignment deals better than hard-assignment with situations where the distribution of training descriptors differs significantly from the testing data.

Figure 5 shows some matching examples with and without soft-assignment, including spatial verification. Soft-assignment generally increases the number of inliers and the quality of the object localization in the target image. This helps to reduce confusion and boost performance.

Figure 4 shows precision-recall curves for two queries

	SP	QE	Testing D1 Training data Oxford	Testing D1 Training data Paris	Testing D1+D2 Training data Oxford	Testing D1+D2 Training data Paris
Hard			0.614	0.403	0.498	0.290
Hard	×		0.653	0.460	0.565	0.385
Hard	×	×	0.801	0.654	0.708	0.562
Soft			0.673	0.493	0.534	0.343
Soft	×		0.731	0.598	0.620	0.480
Soft	×	×	**0.825**	**0.718**	**0.719**	**0.605**

Table 5. Performance of hard- and soft-assigned vocabularies with spatial verification (SP) and query expansion (QE). The performance is evaluated on the *Oxford* dataset with vocabularies built from *Oxford* or *Paris* datasets. These results are for a 1M vocabulary.

from the Oxford dataset. In both cases, the gain in performance is large (`ashmolean_3` goes from 0.626 AP to 0.874 AP, `christ_church_5` increases from 0.333 to 0.813 AP).

Scaling-up to 100K images: Here we evaluate benefits of soft-assignment when the testing database is scaled-up to more than 100K images (D2 dataset). Results including the spatial verification and query expansion are summarized in table 5. It is evident that using soft-assignment always boosts performance over using hard assignments, though the difference is not as pronounced as for the D1 dataset.

6.2. Discussion

Soft assignment brings a performance boost over the original hard assignment method – indeed the descriptor-space soft assignment boosts every stage of the processing: original returns, after spatial verification, and after query expansion. It also offers some protection against a change in vocabulary.

Before spatial verification, it seems that soft assignment predominantly boosts recall so that the spatial verification is able to increase precision over much more of the dataset. This is the principal reason that the technique boosts performance when using query expansion.

It is worth noting that HKM with hierarchical scoring [14] is also a version of soft assignment since, for example, scoring with one non-leaf level in a tree with branching factor ten is an approximation of ten nearest neighbor soft assignment. However, as demonstrated by the results in table 3, it does not perform even as well as hard assignment. There are several reasons for this: first, the HKM clustering method introduces decision boundaries on each level of the tree, magnifying the quantization effects. Second, the weights (relevance of each visual word) in this case are not governed by the distance of the descriptor to the cluster centers (but by the ten cluster centers associated with the non-leaf node).

7. Image-space soft assignment

We also conducted experiments to address the problems of quantization which are not local in the SIFT space. As an example consider the orientation instability of SIFT. To understand this instability we have to divide the SIFT extraction into three steps: affine normalization up to rotation, rotation estimation, and Euclidean description. The first step is implemented as normalization of an ellipse to a circle, the second as a dominant orientation detection, and the last step computes the final weighted histograms of gradient orientations. Noise changes the affine normalization and histogram steps in a continuous way, but a change in the selection of the dominant orientation (a change of mode) results in a discontinuous change in the final descriptor. The result is a non-local change in SIFT space, and consequently the assigned visual words are not neighbours.

In order to address this type of non-local change, we simulated possible image degradations including affine perturbation of the image as well as additive noise. A similar strategy has been used in [10] in learning classifiers for robust pose estimation. Hessian affine features were then detected in each image. These features were described by a SIFT descriptor and assigned to an appropriate visual word. The final image descriptor was an average over 50 perturbations.

Compared to descriptor-space soft-assignment, image-space soft assignment is much more computationally expensive to generate (as new image features need to be computed), but in principle can estimate complex distributions beyond those determined by isotropic Gaussian weighting amongst nearest neighbors in descriptor-space – for example the distribution may be anisotropic – as well as non-local *wormholes* to entirely different regions of the descriptor space.

In our experiments, image-space soft assignment had inferior mAP performance to the descriptor-space method, at higher computational cost. The performance of the image-space method, trained and tested on the D1 dataset was 0.640 mAP with spatial re-ranking turned off, and 0.746 mAP with query expansion. Despite the performance gap of this approach to the descriptor-based method, the intuition behind soft-assignment of features that are not adjacent in descriptor space may still be valid, and the poor performance could have been due to our particular choices of image perturbation. We believe this technique may reward further research.

8. Conclusion

A new method of visual word assignment was introduced: descriptor-space soft-assignment. It improves the state of the art performance on standard datasets by collecting information about image patches that is lost in the quantization step of previously published methods. The performance over using hard-assignment is outstanding when no relevance feedback techniques are used, but the advantage is less pronounced when the method is used in combination with query expansion. This is not surprising, since the query expansion leaves less space for improvement. However, soft-assignment is still worthwhile even in the case of query expansion and, in particular, when the vocabulary is learnt on a different dataset to the one being tested.

Acknowledgements. We are grateful for support from an EPSRC Platform grant, the Royal Academy of Engineering, EC grant 215078 DIPLECS, and Microsoft.

References

[1] http://www.robots.ox.ac.uk/∼vgg/data/oxbuildings/.

[2] http://www.robots.ox.ac.uk/∼vgg/data/parisbuildings/.

[3] Y. Aasheim, M. Lidal, and K. Risvik. Multi-tier architecture for web search engines. *Web Congress, 2003. Proceedings. First Latin American*.

[4] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, ISBN: 020139829, 1999.

[5] L. Barroso, J. Dean, and U. Holzle. Web search for a planet: The google cluster architecture. *Micro, IEEE*, 23, 2003.

[6] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[7] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *Proc. ICCV*, 2007.

[8] A. Gersho and R. Gray. *Vector quantization and signal compression*. Kluwer Academic Publishers, Boston, 1992.

[9] H. Jegou, H. Harzallah, and C. Schmid. A contextual dissimilarity measure for accurate and efficient image search. In *Proc. CVPR*, 2007.

[10] V. Lepetit, J. Pilet, and P. Fua. Point matching as a classification problem for fast and robust object pose estimation. In *Proc. CVPR*, 2004.

[11] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[12] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *Proc. BMVC.*, pages 384–393, 2002.

[13] K. Mikolajczyk and C. Schmid. Scale & affine invariant interest point detectors. *IJCV*, 1(60):63–86, 2004.

[14] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proc. CVPR*, 2006.

[15] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proc. CVPR*, 2007.

[16] C. Silpa-Anan and R. Hartley. Localization using an image-map. In *Australasian Conf. on Robotics and Automation*, 2004.

[17] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proc. ICCV*, 2003.

[18] T. Tuytelaars and C. Schmid. Vector quantizing feature space with a regular lattice. In *Proc. ICCV*, 2007.